Counting spanning trees in self-similar networks by evaluating determinants

Yuan Lin, Bin Wu, and Zhongzhi Zhang*
School of Computer Science, Fudan University, Shanghai 200433, China and
Shanghai Key Lab of Intelligent Information Processing, Fudan University, Shanghai 200433, China

Guanrong Chen†
Department of Electronic Engineering, City University of Hong Kong, Hong Kong, China

(Dated: January 20, 2013)

Spanning trees are relevant to various aspects of networks. Generally, the number of spanning trees in a network can be obtained by computing a related determinant of the Laplacian matrix of the network. However, for a large generic network, evaluating the relevant determinant is computationally intractable. In this paper, we develop a fairly generic technique for computing determinants corresponding to self-similar networks, thereby providing a method to determine the numbers of spanning trees in networks exhibiting self-similarity. We describe the computation process with a family of networks, called (x, y)-flowers, which display rich behavior as observed in a large variety of real systems. The enumeration of spanning trees is based on the relationship between the determinants of submatrices of the Laplacian matrix corresponding to the (x, y)-flowers at different generations and is devoid of the direct laborious computation of determinants. Using the proposed method, we derive analytically the exact number of spanning trees in the (x, y)-flowers, on the basis of which we also obtain the entropies of the spanning trees in these networks. Moreover, to illustrate the universality of our technique, we apply it to some other self-similar networks with distinct degree distributions, and obtain explicit solutions to the numbers of spanning trees and their entropies. Finally, we compare our results for networks with the same average degree but different structural properties, such as degree distribution and fractal dimension, and uncover the effect of these topological features on the number of spanning trees.

PACS numbers: 89.75.Hc, 05.50.+q, 05.20.-y, 04.20.Jb

I. INTRODUCTION

Counting spanning trees in networks is a fundamental issue in both theory and applications, which has recently attracted increasing attention from mathematics [1–4], physics [5–10], and other fields [11]. As a network invariant, spanning tree is a crucial measure of the network reliability [12–14]. The notion is also closely related to various aspects of networks. For example, the number of spanning trees of a connected network is exactly the number of recurrent configurations of the Abelian sandpile model on the network [15, 16], which is equivalent to the chip-firing game [17] under certain restrictions and is a paradigm for self-organized criticality [18]. As another example, spanning trees offer useful insights into understanding the origin and mechanism of fractality in fractal scale-free networks [19]. In addition, spanning trees are relevant to other interesting problems on networks, such as transport [20], electrical networks [21], unbiased random walks [22, 23], loop-erased random walks [24], and so on.

In view of their relevance to diverse aspects of networks and a wide range of applications [25], spanning trees in networks have become a focus of some recent research [26–32]. Particularly, in the physics literature a lot of efforts have been devoted to enumerating spanning trees in specific networks by using different techniques according to their special structures. Examples include regular lattices [5–7, 9], Sierpinski gaskets [8, 10], Erdsö-Rényi random graphs [4], the pseudofractal scale-free web [33], fractal scale-free networks [34], and so on. These works have provided some effective methods for determining spanning trees in special networks, and have revealed some nontrivial effects of network structural properties on spanning trees.

Most existing methods for counting the number of spanning trees are applicable only to particular networks. Due to the complexity and diversity of networks, a general approach for counting spanning trees of a generic network is not available, with the exception of the Kirchhoff Matrix Tree Theorem [35]. This well-known theorem provides a universal algorithm for determining the number of spanning trees of an arbitrary connected graph, in terms of a determinant [36]. However, the computational complexity for evaluating the determinant of a general network is very high [37]. For large networks, it is difficult and even impossible to obtain their numbers of spanning trees. For this reason, seeking an efficient and somewhat general technique for calculating the determinant of a network is a matter of exceptional importance.

On the other hand, it has been observed that by nature most real networks are fractal [38] as well as scale-free [39]. Both striking properties have significant influences on other structural properties, e.g., average dis-

*Electronic address: zhangzz@fudan.edu.cn; URL: http://homepage.fudan.edu.cn/~zhangzz/
†Electronic address: eegchen@cityu.edu.hk

tance [40–42] and degree correlations [43]. In particular, existing results have shown that in some special networks, the scale-free property may decrease the number of spanning trees [33], while fractality can increase the number of spanning trees [34]. Thus, some interesting questions rise naturally: Can the power-law behavior alone determine the number of spanning trees in a scale-free network? What is the relation between the fractal dimension and the spanning trees in a fractal scale-free network? Both questions remain unsolved today.

In this paper, we present an approach for explicitly determining the number of spanning trees by evaluating a related determinant in a network, which is expected to hold true for a broad class of self-similar graphs. Our method is based on the recursion relations of determinants of submatrices of the Laplacian matrix associated with the network at different iterations, which avoids the laborious computation of the largest determinant as needed by classic algebraic schemes. To the best of our knowledge, this constitutes the first work in exploring fast evaluation of the determinant of a complex network, especially for networks with scale-free [39], small-world [44], and fractal [38] properties.

To demonstrate the computation process of our technique, we apply it to find the number of spanning trees in the family of self-similar networks, called (x, y)-flowers [45, 46], which display some remarkable properties observed in real-life networks [47–49] and include the extensively studied pseudofractal scale-free web [50, 51] and fractal hierarchical lattices [52–55] as their limiting cases. Using the proposed technique, we derive the exact number of spanning trees in the whole family of (x, y)-flowers, based on which we obtain their entropies of spanning trees. In addition, in order to exhibit the generality of our method, we apply it to some other self-similar networks and obtain closed-form solutions for the numbers of spanning trees in such networks. Finally, we present a detailed analysis of the obtained results, and show that power-law distribution alone do not suffice to characterize the numbers of spanning trees in scale-free networks, and that the increasing fractal dimensions of fractal scale-free networks may lead to decrements of the numbers of spanning trees.

II. MODEL AND PROPERTIES OF (x, y)-FLOWERS

In this section, we introduce the construction and structural properties of the (x, y)-flowers [45, 46], which are built in an iterative way. Let $F_n(x, y)$ $(n \geq 0)$ denote the (x, y)-flowers after n generations of evolution. In what follows it is assumed that $x \leq y$ and $y > 1$ without loss of generality. The (x, y)-flowers are constructed as follows. For $n = 0$, $F_0(x, y)$ is an edge connecting two nodes, called the initial nodes hereafter. For $n \geq 1$, $F_n(x, y)$ is derived from $F_{n-1}(x, y)$ through replacing each exiting edge in $F_{n-1}(x, y)$ by two parallel paths of lengths x and y, see Fig. 1. For illustration, in Figs. 2 and 3, we present the growing processes of two particular networks: $(1, 3)$-flower and $(2, 2)$-flower. According to the generating algorithm for the (x, y)-flowers, it is easy to see that the number of edges in $F_n(x, y)$ is

$$M_n = (x + y)^n\,. \tag{1}$$

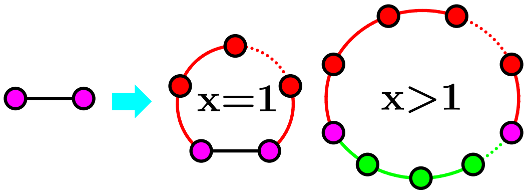

FIG. 1: (Color online) Iterative construction method of the (x, y)-flowers. Each edge is replaced by two parallel paths with lengths x $(x \geq 1)$ and y $(y \geq x$ and $y > 1)$ on the right-hand side of the arrow. For $x = 1$, a pair of old nodes directly connected by an old edge generates $y - 1$ new nodes. All of these $y - 1$ new nodes and the two old nodes form a red path of length y; while the old edge is remained. For $x > 1$, each old edge connecting two old nodes is removed and replaced by two paths with the two old nodes as the ends of the two paths: the $y - 1$ red nodes and the two old nodes form a path of length y, while the $x - 1$ green nodes and the two old nodes constitute another path of length x.

Alternatively, the (x, y)-flowers can be constructed by using the following equivalent algorithm [45], which highlights the self-similarity of the networks. From $F_{n-1}(x, y)$, the next generation $F_n(x, y)$ is obtained by joining $x + y$ copies of $F_{n-1}(x, y)$ at the two initial nodes of each replica. Figure 4 provides an illustration for the second construction approach for the $(1, 3)$-flower. According to this construction, the number of nodes N_n in $F_n(x, y)$ satisfies the recursive relation $N_n =$

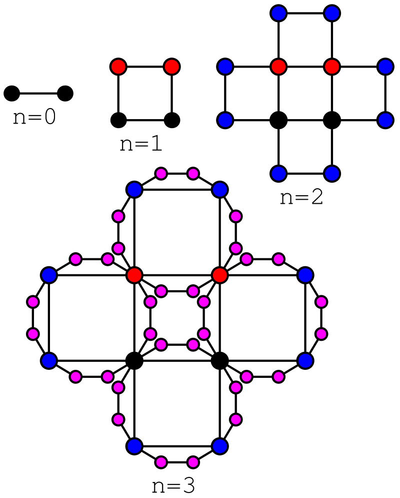

FIG. 2: (Color online) Illustration of the growing process for the $(1, 3)$-flower.

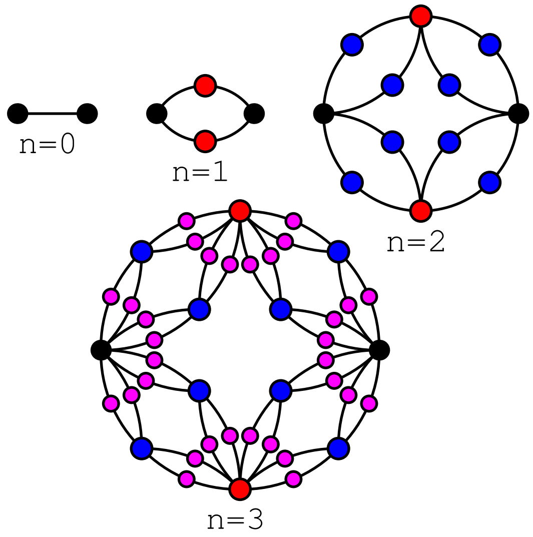

FIG. 3: (Color online) Sketch of the iterative process for the $(2,2)$-flower.

$(x+y)N_{n-1}-(x+y)$, which together with the initial condition $N_1 = x+y$ yields

$$N_n = \frac{x+y-2}{x+y-1}(x+y)^n + \frac{x+y}{x+y-1}\,. \tag{2}$$

Thus, the average degree of the (x,y)-flowers is

$$\langle k\rangle_n = \frac{2M_n}{N_n} = \frac{2(x+y-1)(x+y)^{n-1}}{(x+y-2)(x+y)^{n-1}+1}\,, \tag{3}$$

which approaches $2(x+y-1)/(x+y-2)$ as $n \to \infty$.

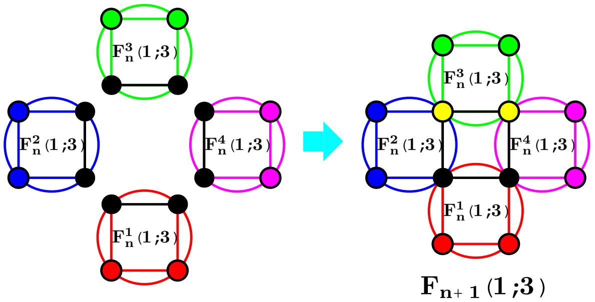

FIG. 4: (Color online) Illustration of the alternative construction algorithm for the $(1,3)$-flower. $F_{n+1}(1,3)$ is obtained through amalgamating four copies of $F_n(1,3)$, denoted by $F_n^\eta(1,3)$ ($\eta = 1,2,3,4$), by merging four pairs of the initial nodes belonging to different $F_n^\eta(1,3)$. Black nodes represent the initial nodes of $F_n^\eta(1,3)$ or of $F_{n+1}(1,3)$.

The deterministic construction also allows to determine the full degree distribution of the (x,y)-flowers. In $F_n(x,y)$, the degree spectra of all nodes are discrete, i.e., nodes have only degree k in the form of $k = 2^m$ ($m = 1,2,\dots,n$). Let $N_n(m)$ be the number of nodes with degree 2^m in $F_n(x,y)$. Then, $N_n(m) = (x+y-2)(x+y)^{n-m}$ for $m < n$, and $N_n(m) = x+y$ for $m = n$. Thus, in the family of $F_n(x,y)$, all networks with the same $x+y$ have an identical degree sequence, hence an identical degree distribution.

The (x,y)-flowers exhibit rich behavior in their architecture [45, 46]. They obey a power-law degree distribution $P(k) \sim k^{-\gamma}$ with the exponent $\gamma = 1+\ln(x+y)/\ln 2$ lying in the interval $[1+\ln 3/\ln 2, \infty)$. For $x = 1$, the networks are small-world but non-fractal; while for $x > 1$, they are "large-world" and fractal with the fractal dimension $d_{\rm f} = \ln(x+y)/\ln x$. These topological properties are not shared by any other modeling networks, which make the (x,y)-flowers unique within the family of networks.

III. SPANNING TREES IN (x,y)-FLOWERS

After introducing the construction and some characteristics of the (x,y)-flowers, we now attempt to determine analytically the number of spanning trees in $F_n(x,y)$, denoted by $N_{\rm ST}(n)$. According to the Matrix Tree Theorem [35, 36], $N_{\rm ST}(n)$ can be evaluated by computing the determinant of a submatrix of the Laplacian matrix corresponding to $F_n(x,y)$, which is obtained by removing a row and a column corresponding to an arbitrary node of $F_n(x,y)$. The elements of the Laplacian matrix associated with $F_n(x,y)$, represented by F_n, are defined as follows: the non-diagonal entry f_{ij} ($i \neq j$) is -1 (or 0) if nodes i and j are (or not) directly connected by an edge, while the diagonal entry f_{ii} equals the degree of node i.

To find $N_{\rm ST}(n)$, we denote by G_n the submatrix of F_n obtained by removing from F_n the row and column corresponding to an initial node of $F_n(x,y)$. Then, we have

$$N_{\rm ST}(n) = \det(G_n). \tag{4}$$

Our aim is to determine $\det(G_n)$. Although for a general network of large size, the evaluation of its determinant is very hard and even impossible, below we show that for $F_n(x,y)$ and other self-similar networks, we can evaluate the associated determinants and derive closed-form expressions for the number of their spanning trees.

First, we define another submatrix of F_n, which is obtained by deleting the two rows and columns from F_n corresponding to the two initial nodes, and denote it by H_n. In what follows, we show the recursive relations for $\det(G_n)$ and $\det(H_n)$, thereby, expressing $\det(G_{n+1})$ and $\det(H_{n+1})$ in terms of $\det(G_n)$ and $\det(H_n)$, based on which we offer explicit solutions for these two quantities. For simplicity, let f_n, g_n, and h_n represent $\det(F_n)$, $\det(G_n)$, and $\det(H_n)$, respectively. Obviously, $f_n = \det(F_n) = 0$ for all n, since the sum of each row in F_n is zero.

It should be noted that the technique we develop for evaluating the determinants can be adapted to a wide class of self-similar networks other than the (x,y)-flowers. Here, we illustrate the computation process with the (x,y)-flowers. Moreover, we only consider in detail two particular and representative networks in the (x,y)-flower family, namely, $(1,3)$-flower and $(2,2)$-flower. For the general (x,y)-flowers we can treat them analogously although the computation is somewhat complex. In so doing, for notational simplicity, we only provide the recursion relations governing g_n and h_n for the general (x,y)-flowers, but omit their detail computations.

A. The $(1,3)$-flower

According to the second construction algorithm of the networks, see Fig. 4, the Laplacian matrix of $F_{n+1}(1,3)$, denoted by F_{n+1} in the case without confusion, can be expressed in terms of H_n as

$$F_{n+1} = \begin{pmatrix} 2d_n & -1 & 0 & -1 & p_n & 0 & 0 & q_n \\ -1 & 2d_n & -1 & 0 & q_n & p_n & 0 & 0 \\ 0 & -1 & 2d_n & -1 & 0 & q_n & p_n & 0 \\ -1 & 0 & -1 & 2d_n & 0 & 0 & q_n & p_n \\ p_n^\top & q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ 0^\top & p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ q_n^\top & 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{pmatrix}. \quad (5)$$

In Eq. (5), the first two rows and two columns correspond to the two initial nodes of $F_{n+1}(1,3)$; d_n is the degree of either of initial nodes at generation n; p_n and q_n are two row vectors with order $N_n - 2 = \frac{2}{3}(4^n - 1)$ describing respectively the interaction between the two initial nodes and the other nodes in different replicas of $F_n(1,3)$ that are joined together forming $F_{n+1}(1,3)$; and the superscript $\top$ of a vector represents its transpose. The third and forth rows and columns describe the connection relation of the two nodes created at $n = 1$ (each of which has the same degree $2d_n$ as the two initial nodes) and other nodes in $F_{n+1}(1,3)$.

Based on Eq. (5), we can also represent G_{n+1} and H_{n+1} in terms of H_n, as

$$G_{n+1} = \begin{pmatrix} 2d_n & -1 & 0 & q_n & p_n & 0 & 0 \\ -1 & 2d_n & -1 & 0 & q_n & p_n & 0 \\ 0 & -1 & 2d_n & 0 & 0 & q_n & p_n \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{pmatrix} \quad (6)$$

and

$$H_{n+1} = \begin{pmatrix} 2d_n & -1 & 0 & q_n & p_n & 0 \\ -1 & 2d_n & 0 & 0 & q_n & p_n \\ 0^\top & 0^\top & H_n & O & O & O \\ q_n^\top & 0^\top & O & H_n & O & O \\ p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & p_n^\top & O & O & O & H_n \end{pmatrix}. \quad (7)$$

In the case of $n = 2$, for example, Eqs. (6) and (7) become respectively

$$G_2 = \begin{pmatrix} 4 & -1 & 0 & q_1 & p_1 & 0 & 0 \\ -1 & 4 & -1 & 0 & q_1 & p_1 & 0 \\ 0 & -1 & 4 & 0 & 0 & q_1 & p_1 \\ q_1^\top & 0^\top & 0^\top & H_1 & O & O & O \\ p_1^\top & q_1^\top & 0^\top & O & H_1 & O & O \\ 0^\top & p_1^\top & q_1^\top & O & O & H_1 & O \\ 0^\top & 0^\top & p_1^\top & O & O & O & H_1 \end{pmatrix}$$

and

$$H_2 = \begin{pmatrix} 4 & -1 & 0 & q_1 & p_1 & 0 \\ -1 & 4 & 0 & 0 & q_1 & p_1 \\ 0^\top & 0^\top & H_1 & O & O & O \\ q_1^\top & 0_1^\top & O & H_1 & O & O \\ p_1^\top & q_1^\top & O & O & H_1 & O \\ 0^\top & p_1^\top & O & O & O & H_1 \end{pmatrix},$$

where $p_1 = \begin{pmatrix} 0 & -1 \end{pmatrix}$, $q_1 = \begin{pmatrix} -1 & 0 \end{pmatrix}$, and $H_1 = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$.

With the expressions for G_{n+1} and H_{n+1} given in Eqs. (6) and (7), we can evaluate the determinants of G_n and H_n, namely, g_n and h_n, on the basis of their recursive relations that will be derived. We first show a detailed computation process for g_n, while the evaluation for h_n can be dealt with in an analogous way.

Note that the first row of G_{n+1} can be viewed as the sum, i.e., a linear combination with scalars being 1, of two vectors; that is,

$$\begin{aligned} & \begin{pmatrix} 2d_n & -1 & 0 & q_n & p_n & 0 & 0 \end{pmatrix} \\ = & \begin{pmatrix} d_n & 0 & 0 & q_n & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} d_n & -1 & 0 & 0 & p_n & 0 & 0 \end{pmatrix}. \end{aligned}$$

Analogously, the second and third rows can also be regarded separately as the sum of two vectors, namely,

$$\begin{aligned} & \begin{pmatrix} -1 & 2d_n & -1 & 0 & q_n & p_n & 0 \end{pmatrix} \\ = & \begin{pmatrix} -1 & d_n & 0 & 0 & q_n & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & d_n & -1 & 0 & 0 & p_n & 0 \end{pmatrix}, \end{aligned}$$

and

$$\begin{aligned} & \begin{pmatrix} 0 & -1 & 2d_n & 0 & 0 & q_n & p_n \end{pmatrix} \\ = & \begin{pmatrix} 0 & -1 & d_n & 0 & 0 & q_n & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & d_n & 0 & 0 & 0 & p_n \end{pmatrix}. \end{aligned}$$

Thus, according to the properties of determinants, g_{n+1} can be decomposed as the sum of eight determinants, as follows:

$$
\begin{aligned}
g_{n+1} \;=\; & \begin{vmatrix} d_n & 0 & 0 & q_n & 0 & 0 & 0 \\ -1 & d_n & 0 & 0 & q_n & 0 & 0 \\ 0 & -1 & d_n & 0 & 0 & q_n & 0 \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & 0 & 0 & q_n & 0 & 0 & 0 \\ -1 & d_n & 0 & 0 & q_n & 0 & 0 \\ 0 & 0 & d_n & 0 & 0 & 0 & p_n \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & 0 & 0 & q_n & 0 & 0 & 0 \\ 0 & d_n & -1 & 0 & 0 & p_n & 0 \\ 0 & -1 & d_n & 0 & 0 & q_n & 0 \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix} \\
& + \begin{vmatrix} d_n & 0 & 0 & q_n & 0 & 0 & 0 \\ 0 & d_n & -1 & 0 & 0 & p_n & 0 \\ 0 & 0 & d_n & 0 & 0 & 0 & p_n \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & -1 & 0 & 0 & p_n & 0 & 0 \\ -1 & d_n & 0 & 0 & q_n & 0 & 0 \\ 0 & -1 & d_n & 0 & 0 & q_n & 0 \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & -1 & 0 & 0 & p_n & 0 & 0 \\ -1 & d_n & 0 & 0 & q_n & 0 & 0 \\ 0 & 0 & d_n & 0 & 0 & 0 & p_n \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix} \\
& + \begin{vmatrix} d_n & -1 & 0 & 0 & p_n & 0 & 0 \\ 0 & d_n & -1 & 0 & 0 & p_n & 0 \\ 0 & -1 & d_n & 0 & 0 & q_n & 0 \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & -1 & 0 & 0 & p_n & 0 & 0 \\ 0 & d_n & -1 & 0 & 0 & p_n & 0 \\ 0 & 0 & d_n & 0 & 0 & 0 & p_n \\ q_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ p_n^\top & q_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & p_n^\top & q_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & p_n^\top & O & O & O & H_n \end{vmatrix}.
\end{aligned}
$$

Therefore, the evaluation of g_{n+1} is reduced to finding the values of the eight determinants on the right-hand side of Eq. (8), which are denoted sequentially as $g_{n+1}^{(i)}$ $(1 \leq i \leq 8)$. Using some elementary matrix operations, we can verify that $g_{n+1}^{(1)} = g_{n+1}^{(2)} = g_{n+1}^{(4)} = g_{n+1}^{(8)}$ and $g_{n+1}^{(3)} = g_{n+1}^{(5)} = g_{n+1}^{(6)} = g_{n+1}^{(7)}$. Having g_{n+1} in terms of the quantities of $g_{n+1}^{(i)}$, the next step is to explicitly determine these quantities. In view of the equivalence of some $g_{n+1}^{(i)}$, we only compute $g_{n+1}^{(2)}$ and $g_{n+1}^{(3)}$.

For $g_{n+1}^{(2)}$ and $g_{n+1}^{(3)}$, utilizing some elementary matrix operations, we can easily obtain

$$
g_{n+1}^{(2)} = h_n \begin{vmatrix} d_n & q_n & 0 & \mathit{0} & 0 & \mathit{0} \\ q_n^\top & H_n & \mathit{0}^\top & O & \mathit{0}^\top & O \\ -1 & \mathit{0} & d_n & q_n & 0 & \mathit{0} \\ p_n^\top & O & q_n^\top & H_n & \mathit{0}^\top & O \\ 0 & \mathit{0} & 0 & \mathit{0} & d_n & p_n \\ \mathit{0}^\top & O & \mathit{0}^\top & O & p_n^\top & H_n \end{vmatrix} = h_n(g_n)^3
$$

and

$$
\begin{aligned}
g_{n+1}^{(3)} \;=\; & (h_n)^2 \begin{vmatrix} d_n & -1 & p_n & 0 & \mathit{0} \\ -1 & d_n & q_n & 0 & \mathit{0} \\ p_n^\top & q_n^\top & H_n & \mathit{0}^\top & O \\ 0 & 0 & \mathit{0} & d_n & q_n \\ \mathit{0}^\top & \mathit{0}^\top & O & q_n^\top & H_n \end{vmatrix} \\
=\; & (h_n)^2 f_n g_n = 0.
\end{aligned}
$$

Plugging these expressions into Eq. (8), we have

$$
g_{n+1} = \sum_{i=1}^{8} g_{n+1}^{(i)} = 4h_n(g_n)^3. \tag{8}
$$

Next, we derive the recursion relation for h_{n+1}. Through a computational process similar to that for g_{n+1}, we obtain the following relation:

$$
\begin{aligned}
h_{n+1} \;=\; & h_n \left(\begin{vmatrix} d_n & 0 & q_n & \mathit{0} & \mathit{0} \\ -1 & d_n & \mathit{0} & q_n & \mathit{0} \\ q_n^\top & \mathit{0}^\top & H_n & O & O \\ p_n^\top & l_n^\top & O & H_n & O \\ \mathit{0}^\top & p_n^\top & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & 0 & q_n & \mathit{0} & \mathit{0} \\ 0 & d_n & \mathit{0} & \mathit{0} & p_n \\ q_n^\top & \mathit{0}^\top & H_n & O & O \\ p_n^\top & q_n^\top & O & H_n & O \\ \mathit{0}^\top & p_n^\top & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & -1 & \mathit{0} & p_n & \mathit{0} \\ -1 & d_n & \mathit{0} & q_n & \mathit{0} \\ q_n^\top & \mathit{0}^\top & H_n & O & O \\ p_n^\top & q_n^\top & O & H_n & O \\ \mathit{0}^\top & p_n^\top & O & O & H_n \end{vmatrix}
+ \begin{vmatrix} d_n & -1 & \mathit{0} & p_n & \mathit{0} \\ 0 & d_n & \mathit{0} & \mathit{0} & p_n \\ q_n^\top & \mathit{0}^\top & H_n & O & O \\ p_n^\top & q_n^\top & O & H_n & O \\ \mathit{0}^\top & p_n^\top & O & O & H_n \end{vmatrix} \right) \\
=\; & 3(h_n)^2(g_n)^2 + (h_n)^3 f_n = 3(h_n)^2(g_n)^2.
\end{aligned}
$$

Having obtained the above recursive expressions for g_{n+1} and h_{n+1}, we are now in a position to explicitly determine g_n. For this purpose, we examine a new quantity g_n/h_n, which obviously satisfies the following recursive relation:

$$\frac{g_{n+1}}{h_{n+1}} = \frac{4}{3}\frac{g_n}{h_n}.$$

Considering the initial condition $g_1/h_1 = 4/3$, the above equation is solved by induction to yield

$$\frac{g_n}{h_n} = \left(\frac{4}{3}\right)^n.$$

Hence,

$$h_n = \left(\frac{3}{4}\right)^n g_n. \tag{9}$$

Substituting Eq. (9) into Eq. (8) and using $g_1 = 4$, we can solve Eq. (8) to obtain the exact number of spanning trees:

$$N_{\rm ST}(n) = g_n = 3^{(4^n-3n-1)/9}4^{(2\times 4^n+3n-2)/9}. \tag{10}$$

It is easy to express $N_{\rm ST}(n)$ in terms of the network size N_n, so as to obtain the relation governing the two quantities. Recalling that for $(1,3)-$flower, $N_n = (2\times 4^n + 4)/3$, hence we have $4^n = (3N_n - 4)/2$ and $n = [\ln(3N_n - 4)/\ln 2 - 1]/2$. These relations allow us to recast $N_{\rm ST}(n)$ as a function of N_n:

$$N_{\rm ST}(n) = 3^{[N_n - \ln(3N_n-4)/\ln 2-1]/6}4^{[2N_n+\ln(3N_n-4)/\ln 2-5]/6},$$

which indicates that $N_{\rm ST}(n)$ approximately increases exponentially in N_n.

Then we can define the entropy of spanning trees for the $(1,3)-$flower as the limiting value [1, 3, 4]:

$$E_{(1,3)-\text{flower}} = \lim_{N_n\to\infty}\frac{\ln N_{\rm ST(n)}}{N_n} = \frac{1}{6}(4\ln 2+\ln 3) \approx 0.6452.$$

B. The $(2,2)$-flower

We proceed to determine the number of spanning trees in the $(2,2)$-flower. In the computation process, we use the same notations as those defined for the $(1,3)$-flower. Similarly to the $(1,3)$-flower, we can derive the recursion relations for G_{n+1} and H_{n+1}, as

$$G_{n+1} = \begin{pmatrix} 2d_n & 0 & 0 & s_n & r_n & 0 & 0 \\ 0 & 2d_n & 0 & 0 & s_n & r_n & 0 \\ 0 & 0 & 2d_n & 0 & 0 & s_n & r_n \\ s_n^\top & 0^\top & 0^\top & H_n & O & O & O \\ r_n^\top & s_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & r_n^\top & s_n^\top & O & O & H_n & O \\ 0^\top & 0^\top & r_n^\top & O & O & O & H_n \end{pmatrix}, \tag{11}$$

and

$$H_{n+1} = \begin{pmatrix} 2d_n & 0 & s_n & r_n & 0 & 0 \\ 0 & 2d_n & 0 & 0 & s_n & r_n \\ s_n^\top & 0^\top & H_n & O & O & O \\ r_n^\top & 0^\top & O & H_n & O & O \\ 0^\top & s_n^\top & O & O & H_n & O \\ 0^\top & r_n^\top & O & O & O & H_n \end{pmatrix}, \tag{12}$$

respectively. In Eqs. (11) and (12), r_n and s_n are two vectors describing the connection relation between an initial node and the other nodes.

Based on the above two relations, we obtain the following relations for the determinants corresponding to G_{n+1} and H_{n+1} by using some elementary matrix operations:

$$g_{n+1} = 4h_n(g_n)^3 + 4(h_n)^2 f_n g_n = 4h_n(g_n)^3,$$

and

$$h_{n+1} = 4(h_n)^2(g_n)^2.$$

These two recursion relations, coupled with the initial conditions $g_1 = 4$ and $h_1 = 4$, yield

$$N_{\rm ST}(n) = g_n = 2^{\frac{2}{3}(4^n-1)}, \tag{13}$$

which can be further expressed in terms of the network size N_n as

$$N_{\rm ST}(n) = 2^{N_n-2}.$$

Thus, for the $(2,2)$-flower, the number of spanning trees also scales exponentially with the network size. And its entropy of spanning trees is

$$E_{(2,2)-\text{flower}} = \lim_{N_n\to\infty}\frac{\ln N_{ST}(n)}{N_n} = \ln 2 \approx 0.6931,$$

a value larger than that corresponding to the $(1,3)$-flower.

C. The (x,y)-flowers

The above process for computing spanning trees in the $(1,3)$-flower and the $(2,2)$-flower can be used to determine the number of spanning trees in the whole family of (x,y)-flowers. Since for the general (x,y)-flowers, the recursive relations for G_{n+1} and H_{n+1} are lengthy and the computation process for g_{n+1} and h_{n+1} is more complex, we only provide the recursive relations for g_{n+1} and h_{n+1}, leaving out the derivation detail, which is similar to those for the two particular cases considered above. The recursive relations of g_{n+1} and h_{n+1} for the (x,y)-flowers are given by

$$g_{n+1} = (x+y)h_n(g_n)^{x+y-1} \tag{14}$$

and

$$h_{n+1} = xy(h_n)^2(g_n)^{x+y-2}. \tag{15}$$

Considering the initial conditions $g_1 = x+y$ and $h_1 = xy$, we can solve Eqs. (14) and (15) to obtain the exact number of spanning trees for the (x,y)-flowers:

$$\begin{aligned} N_{\rm ST}(n) &= g_n = (xy)^{\frac{(x+y)^n-(x+y-1)n-1}{(x+y-1)^2}} \\ &\times(x+y)^{\frac{(x+y-2)(x+y)^n+(x+y-1)n-(x+y-2)}{(x+y-1)^2}}. \end{aligned} \tag{16}$$

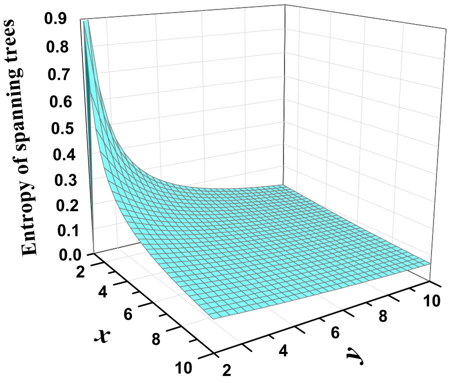

FIG. 5: (Color online) The entropies of spanning trees in the (x, y)-flowers, depending on the two parameters x and y.

For $x = 1$ and $y = 2$, Eq. (16) recovers the result previously obtained in [33]; for $x = 1$ and $y = 3$, Eq. (16) reduces to Eq. (10); while for $x = 2$ and $y = 2$, Eq. (16) agrees with the result given by Eq. (13). The consistency confirms the validity of the formula for spanning trees in the (x, y)-flowers.

Since the size for the (x, y)-flowers is $N_n = (x + y - 2)/(x + y - 1)(x + y)^n + (x + y)/(x + y - 1)$, it is evident from Eq. (16) that $N_{\rm ST}(n)$ scales exponentially with N_n. This permits to get the entropies of spanning trees for the (x, y)-flowers as follows:

$$\begin{aligned} E_{(x,y)-\text{flowers}} &= \lim_{N_n\to\infty} \frac{\ln N_{\rm ST}(n)}{N_n} \\ &= \frac{\ln xy + (x+y-2)\ln(x+y)}{(x+y-1)(x+y-2)}. \end{aligned} \quad (17)$$

It is worthwhile to mention that the above technique and process for finding spanning trees in the (x, y)-flowers by evaluating determinants can be adapted to other self-similar networks. In order to show the universality, in the Appendix we apply the approach to determine the number of spanning trees and their entropies for a small-world network [56] with an exponential degree distribution and a scale-free Apollonian network [57, 58], and obtain explicit solutions to the corresponding quantities.

D. Analysis of the results

The entropy for spanning trees is a very important quantity characterizing the network structure. Thus, it is interesting to show its dependence on the network structure. Since, for the whole family of the (x, y)-flowers, its architecture is controlled by the two parameters x and y, it is expected that the entropy is dependent on x and y. In Fig. 5, we show how the entropy varies with x and y. The intrinsic dependence relation of entropy on x and y is somewhat complex. Here we are mainly concerned about how the entropy $E_{(x,y)-\text{flowers}}$ is reliant on x and y when the sum $x+y$ is fixed with $x+y \geq 4$. The main reason why we are interested in this case lies in that for this case it is meaningful to compare the entropy. As shown in section II, for the (x, y)-flowers with fixed $x + y$, they have an identical average degree. In addition, for fixed $x + y$, the networks have the same degree sequence and thus the same degree distribution. Particularly, for $x > 1$ with fixed $x+y$, the networks are fractal having the fractal dimension monotonously decreasing in x.

Figure 5 shows that when $x + y$ is fixed, $E_{(x,y)-\text{flowers}}$ is determined by the difference between x and y: the larger the difference, the smaller the entropy. Thus, in the case of fixed $x+y$, $E_{(x,y)-\text{flowers}}$ is an increasing function of parameter x. For example, as shown above, when $x + y = 4$, the entropy for the $(1, 3)$-flower is less than that for the $(2, 2)$-flower. Thus, we can conclude that the power-law degree distribution alone can not determine the entropies for spanning trees in scale-free networks. In fact, other than scale-free networks, the degree distributions of other networks do not suffice to characterize the number of spanning trees either. For instance, both the small-world network [56] and the Koch network family [59, 60], to be addressed in the Appendix, have the same average degree and the same entropy of spanning trees, but their degree distributions are disparate: the former is exponential while the latter is scale-free.

As pointed out above, degree distribution alone cannot determine the number of spanning trees. Then an important question arises: which structural property has the decisive effect on the number of spanning tress in the (x, y)-flower family with an identical value of $x + y$? Since, for the whole family of (x, y)-flowers with the same $x + y$, they have the same average node degree, the same degree sequence and thus the same degree distribution, and the same (zero) clustering coefficient, we argue that the difference in the number of spanning trees seen in Fig. 5 is attributed solely to the fractality. Specifically, the fractal property of networks significantly increases the number of spanning trees, which is related to the fractal dimension: the larger the fractal dimension, the less the number of spanning trees. This is obvious from Fig. 5. For those (x, y)-flowers with the same value of $x + y$, when x grows from 1 to its maximum, their fractal dimension $d_{\rm f}$ decreases from ∞ to its minimum, while the entropies of spanning trees increase from their minimum to maximum. Particularly, when $x = 1$, the network is non-fractal, it has the minimum number of spanning trees.

Before closing this section, we discuss the case of different values of $x + y$. Figure 5 shows that for two network classes with different values of $x+y$, the class with smaller $x + y$ has a larger entropy than that of the other class. This can be easily understood. Since for two networks with distinct values of $x + y$, their average node degrees are also disparate: the network with smaller $x + y$ has a larger average node degree than that the other. Thus, the edge density of the former is greater than that of the latter, which is responsible for the difference of entropies for spanning trees in the two networks.

IV. CONCLUSIONS

In this paper, we have developed a rather generic technique for evaluating determinants used to compute the numbers of spanning trees in self-similar networks. Our method is grounded on the recursive relations for the determinants of submatrices of the Laplacian matrix corresponding to the network at two consecutive generations, which are established based on the structure of the network. An advantage of our approach is that it avoids the laborious computation of a determinant and reduces the computational complexity that a standard method needs. We have illustrated our technique by applying it to the family of (x, y)-flowers, which exhibits rich and special structural properties, as well as some remarkable features of many real systems. Moreover, to demonstrate the universality of our approach, we have applied it to some other networks with self-similarity. We obtained explicit formulas for the numbers of spanning trees in different networks. Our results show that degree distribution alone does not suffice to characterize the number of spanning trees in a network, and that fractal dimension has a predominant influence on the number of spanning trees in fractal scale-free networks with the same average degree.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No. 61074119 and by the Hong Kong Research Grants Council under the GRF Grant CityU 1117/10E.

Appendix: Spanning trees in other self-similar networks

To illustrate the university of our proposed method, we apply it to calculate the number of spanning trees in some other self-similar networks with different degree distributions, exponential or scale-free.

1. Small-world exponential network

We first discuss a self-similar small-world network with an exponential form of degree distribution [56], which is observed from some real-life systems [61]. The small-world network is constructed iteratively [56], denoted by S_n after n $(n \geq 0)$ iterations. The network starts from a triangle S_0, with three nodes called initial nodes. For $n \geq 1$, S_n is obtained from S_{n-1}: for each existing node in S_{n-1}, two new nodes are generated, which and their mother node together form a new triangle. Figure 6 shows the structure of S_2. It is easy to derive that the total number of nodes in S_n is $N_n = 3^{n+1}$. The resultant network has a degree distribution decaying exponentially with the degree. In addition, it is small-world with the average distance growing logarithmically with its size.

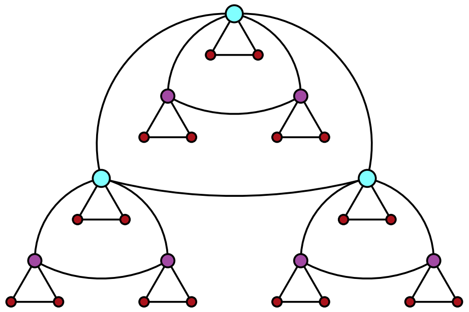

FIG. 6: (Color online) An illustration for the network S_2.

We now begin to evaluate the number of spanning trees in S_n. In the case without any confusion, we use F_n to denote the Laplacian matrix for S_n, and use G_n to denote a submatrix of F_n, which is obtained by removing from F_n the row and column corresponding to one of three initial nodes of S_n. We have

$$F_{n+1} = \begin{pmatrix} d_n+2 & -1 & -1 & u_n & 0 & 0 \\ -1 & d_n+2 & -1 & 0 & u_n & 0 \\ -1 & -1 & d_n+2 & 0 & 0 & u_n \\ u_n^\top & 0^\top & 0^\top & G_n & O & O \\ 0^\top & u_n^\top & 0^\top & O & G_n & O \\ 0^\top & 0^\top & u_n^\top & O & O & G_n \end{pmatrix} \tag{A.1}$$

and

$$G_{n+1} = \begin{pmatrix} d_n+2 & -1 & 0 & u_n & 0 \\ -1 & d_n+2 & 0 & 0 & u_n \\ 0^\top & 0^\top & G_n & O & O \\ u_n^\top & 0^\top & O & G_n & O \\ 0^\top & u_n^\top & O & O & G_n \end{pmatrix}, \tag{A.2}$$

where d_n is the degree of an initial node in S_n, and u_n is a row vector representing the interaction between an initial node and the other non-initial nodes.

According to the Matrix Tree Theorem, the number of spanning trees in S_n is $N_{ST}(n) = \det(G_n)$. Using a similar iterative process as that in the main text, Eq. (A.2) leads to the following relation: $\det(G_n) = 3(\det(G_{n-1}))^3$. Considering the initial condition of $\det(G_0) = 3$, we obtain the exact solution to the number of spanning trees in the small-world network S_n as

$$N_{\rm ST}(n) = 3^{(3^{n+1}-1)/2}, \tag{A.3}$$

and the entropy of spanning trees as

$$E_{S_n} = \lim_{N_n \to \infty} \frac{\ln N_{\rm ST}(n)}{N_n} = \frac{1}{2} \ln 3 \approx 0.5493. \tag{A.4}$$

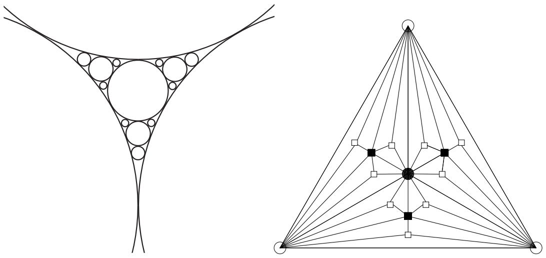

FIG. 7: The classic Apollonian packing of disks and its corresponding Apollonian network.

The above computational process for S_n can also be applied to the scale-free Koch networks [59, 60] with the same average node degree 3 as that of S_n. Interestingly, the spanning tree entropy for the Koch networks is identical to S_n despite the fact that they have totally different types of degree distributions.

2. Small-world scale-free Apollonian network

We apply our technique to determine the spanning trees in the Apollonian network with both small-world and scale-free properties [57, 58]. The Apollonian network is derived from the classic Apollonian packing. This well-known packing problem [62] starts with three mutually touching disks and their interstice is filled by a disk that touches all the three initial disks, forming three smaller interstices to be filled, as shown in the left panel of Fig. 7. Let each disk denote a node located at the center of the corresponding disk, linking the centers of touching disks by lines leads to the Apollonian network, as displayed in the right panel of Fig. 7. The Apollonian network can also be constructed iteratively and has the small-world effects [58]. After n iterations, the size of the Apollonian network is $N_n = \frac{3^n+5}{2}$.

Making use of our technique for evaluating the corresponding determinant, we can determine the number of spanning trees in the Apollonian network. Since the computational process is complicated and the relation equations are lengthy, we only provide the main results. The exact number of spanning trees in the Apollonian network after n iterations is

$$N_{\rm ST}(n) = \frac{1}{4}(3^n + 5^n)^2 3^{(3^n-6n+3)/4} 5^{(3^n-2n-1)/4} \, . \tag{A.5}$$

Consequently, the entropy for spanning trees in the Apollonian network is

$$E_{\rm Apol} = \lim_{N_n \to \infty} \frac{\ln N_{\rm ST}(n)}{N_n} = \frac{1}{2}(\ln 3 + \ln 5) \approx 1.3540 \, , \tag{A.6}$$

which is smaller than 1.9902, the entropy for spanning trees in three-dimensional lattices [7] having the same mean degree 6 as this Apollonian network.

[1] R. Burton and R. Pemantle, Ann. Probab. **21**, 1329 (1993).

[2] A. Z. Broder and E. W. Mayr, J. Algorithms **24**, 171 (1997).

[3] R. Lyons, Combin. Probab. Comput. **14**, 491 (2005).

[4] R. Lyons, R. Peled and O. Schramm, Combin. Probab. Comput. **17**, 711 (2008).

[5] F.-Y. Wu, J. Phys. A **10**, L113 (1977).

[6] R. Shrock and F.-Y. Wu, J. Phys. A **33**, 3881 (2000).

[7] J. L. Felker and R. Lyons, J. Phys. A **36**, 8361 (2003).

[8] S.-C. Chang, L.-C. Chen, and W.-S. Yang, J. Stat. Phys. **126**, 649 (2007).

[9] E. Teufl and S. Wagner, J. Phys. A **43**, 415001 (2010).

[10] E. Teufl and S. Wagner, J. Stat. Phys. **142**, 879 (2011).

[11] R. Jayakumar and K. Thulasiraman. IEEE Trans. Circuits Syst. **36**, 219 (1989).

[12] F. T. Boesch J. Graph Theory **10**, 339 (1986).

[13] G. J. Szabó, M. Alava, and J. Kertész, Physica A **330**, 31 (2003).

[14] F. T. Boesch, A. Satyanarayana, and C. L. Suffel, Networks **54**, 99 (2009).

[15] D. Dhar, Phys. Rev. Lett. **64**, 1613 (1990).

[16] D. Dhar Physica A **369**, 29 (2006).

[17] C. Merino Discrete Math. **302**, 188 (2005).

[18] P. Bak, C. Tang, and K. Wiesenfeld, Phys. Rev. Lett. **59**, 381 (1987).

[19] K.-I. Goh, G. Salvi, B. Kahng and D Kim, Phys. Rev. Lett. **96**, 018701 (2006).

[20] Z. H. Wu, L. A. Braunstein, S. Havlin, and H. E. Stanley, Phys. Rev. Lett. **96**, 148702 (2006).

[21] R. B. Bapat and S. Gupta, Indian J. Pure Appl. Math. **41**, 1 (2010).

[22] J. D. Noh and H. Rieger, Phys. Rev. Lett. **92**, 118701 (2004).

[23] S. Condamin, O. Bénichou, V. Tejedor, R. Voituriez, and J. Klafter, Nature (London) **450**, 77 (2007).

[24] D. Dhar and A. Dhar Phys. Rev. E **55**, 2093(R) (1997).

[25] B. Y. Wu and K.-M. Chao, *Spanning Trees and Optimization Problems* (Chapman & Hall/CRC, Boca Raton, 2004).

[26] R. Dobrin and P. M. Duxbury, Phys. Rev. Lett. **86**, 5076 (2001).

[27] F.-Y. Wu, Int. J. Mod. Phys. B **16**, 1951 (2002).

[28] D.-H. Kim, J. D. Noh, and H. Jeong, Phys. Rev. E **70**, 046126 (2004).

[29] P. J. Macdonald, E. Almaas and A.-L. Barabási, Europhys. Lett. **72**, 308 (2005).

[30] M. A. Serrano, M. Boguna, and A. Vespignani, Proc. Natl. Acad. Sci. U.S.A. **106**, 6483 (2009).

[31] M. Krivelevich, SIAM J. Discrete Math. **24**, 1495 (2010).

[32] K. Ozeki and T. Yamashita, Graphs Combin. **27**, 1 (2011).

[33] Z. Z. Zhang, H. X. Liu, B. Wu, and S. G. Zhou, EPL **90**, 68002 (2010).

[34] Z. Z. Zhang, H. X. Liu, B. Wu, and T. Zou, Phys. Rev. E **83**, 016116 (2011).

[35] G. Kirchhoff, Ann. Phys. Chem. **72**, 497 (1847).

[36] S. Chaiken and D. J. Kleitman, J. Combin. Theory Ser. A **24**, 377 (1978).

[37] N. L. Biggs, *Algebraic Graph Theory*, 2nd ed. (Cambridge University Press, Cambridge, 1993).

[38] C. Song, S. Havlin, H. A. Makse, Nature **433**, 392 (2005).

[39] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).

[40] F. Chung and L. Lu, Proc. Natl. Acad. Sci. U.S.A. **99**, 15879 (2002).

[41] R. Cohen and S. Havlin, Phys. Rev. Lett. **90**, 058701 (2003).

[42] Z. Z. Zhang, S. G. Zhou, T. Zou, L. C. Chen, and J. H. Guan, Phys. Rev. E **79**, 031110 (2009).

[43] C. Song, S. Havlin, H. A. Makse, Nat. Phys. **2**, 275 (2006).

[44] D. J. Watts and H. Strogatz, Nature (London) **393**, 440 (1998).

[45] H. D. Rozenfeld, S. Havlin, and D. ben-Avraham, New J. Phys. **9**, 175 (2007).

[46] H. D. Rozenfeld and D. ben-Avraham, Phys. Rev. E **75**, 061102 (2007).

[47] R. Albert and A.-L. Barabási, Rev. Mod. Phys. **74**, 47 (2002).

[48] S. N. Dorogvtsev and J. F. F. Mendes, Adv. Phys. **51**, 1079 (2002).

[49] M. E. J. Newman, SIAM Rev. **45**, 167 (2003).

[50] S. N. Dorogovtsev, A. V. Goltsev, J. F. F. Mendes, Phys. Rev. E **65**, 066122 (2002).

[51] Z. Z. Zhang, S. G. Zhou, and L. C. Chen, Eur. Phys. J. B **58**, 337 (2007).

[52] A. N. Berker and S. Ostlund, J. Phys. C **12**, 4961 (1979).

[53] M. Kaufman and R. B. Griffiths, Phys. Rev. B **24**, 496 (1981).

[54] R. B. Griffiths and M. Kaufman, Phys. Rev. B **26**, 5022 (1982).

[55] M. Hinczewski and A. N. Berker, Phys. Rev. E **73**, 066126 (2006).

[56] L. Barrière, F. Comellas, C. Dalfó, and M. A. Fiol, Discrete Appl. Math. **157**, 36 (2009).

[57] J. S. Andrade Jr., H. J. Herrmann, R. F. S. Andrade and L. R. da Silva, Phys. Rev. Lett. **94**, 018702 (2005).

[58] Z. Z. Zhang, L. C. Chen, S. G. Zhou, L. J. Fang, J. H. Guan, and T. Zou, Phys. Rev. E **77**, 017102 (2008).

[59] Z. Z. Zhang, S. G. Zhou, W. L. Xie, L. C. Chen, Y. Lin, and J. H. Guan, Phys. Rev. E **79**, 061113 (2009).

[60] Z. Z. Zhang, S. Y. Gao, L. C. Chen, S. G. Zhou, H. J. Zhang, and J. H. Guan, J. Phys. A **43**, 395101 (2010).

[61] L. A. N. Amaral, A. Scala, M. Barthélémy, H. E. Stanley, Proc. Natl. Acad. Sci. U.S.A. **97**, 11149 (2000).

[62] E. Kasner and F. Supnick, Proc. Natl. Acad. Sci. U.S.A. **29**, 378 (1943).